Progress Report & Roadmap CP2

Multicolored Multiplexers

Tyler Jurczyk, Soumil Gupta, Jay Nathan

Progress Report

Functionalities Implemented/Work Completed

For this checkpoint, all the remaining modules were implemented. These include the Dispatcher, Free List, ROB, Reservation Station, Functional Units, Retire RAT, RAT, Physical Register File, CDB. The only remaining parts to code for the base processor is the load/store logic and the branch flushing logic. These functionalities were tested and debugged for a 1-way and 2-way processor (superscalar).

Division of Labor

Our group met multiple times in the past week, mostly in person in the DCL. Division of labor was mainly done by people choosing what they were most comfortable with to code. Lots of time was spent trying to analyze design decisions and through texting, progress updates, decision decisions, and questions were discussed and documented.

Soumil: Reservation Station, Functional Units, Physical Register File, CDB, debugging
Tyler: Dispatcher, Free List, RAT, Top Level, fixing synth issues, debugging
Jay: Retire RAT, Top Level, fixing synth issues, debugging

Testing Strategies

As more modules were written, they were wired up in the top level, and then through verdi they were analyzed to ensure that they held the right values in the right sequence. We went back and forth with debugging for a 1-way and a 2-way processor, and checked the differences in activity per clock cycle for each. A simple program was written with just two instructions to ensure that they could at least go through the pipeline. Once the full pipeline was created, the debugging approach was much closer to the approach in mp_pipeline, where rvfi would raise and issue, and verdi was utilized to trace through the waveforms to identify bugs.

Current Datapath

https://drive.google.com/file/d/1KogmaMYfLVObvLSIMb3sGG44uEM9Bqb_/view?usp=sharing

Roadmap

Planned Progress in Functionalities

Week of 4/1:

1. Add branch logic
2. Add load/store logic (non-speculative for beginning)
3. Debug to get coremark working

Week of 4/8: 411 MT2 Also. Buffer week in the case.

1. EC: Ensure Superscalar works
2. EC: All arithmetic extensions
3. EC: Write next-line prefetching and debug

Week of 4/15:

1. EC: Write Stride Prefetcher and debug
2. EC: Write a Perceptron Branch Predictor, and Debug
3. EC: Write a GShare branch Predictor and Debug

Week of 4/22:

1. EC: Post-Commit Store Buffer
2. EC: Cocotb
3. EC: Non-synthesizable model
4. Write Report
5. Prepare for Presentation

Planned Division of Labor for Next Week

(CP3):

- Branch Logic - Soumil
- LSQ - Tyler, Jay
- Debug with Coremark: All
- Progress Report + Roadmap: Soumil